
LLNL on the BABAR experiment

David J Lange

Douglas Wright

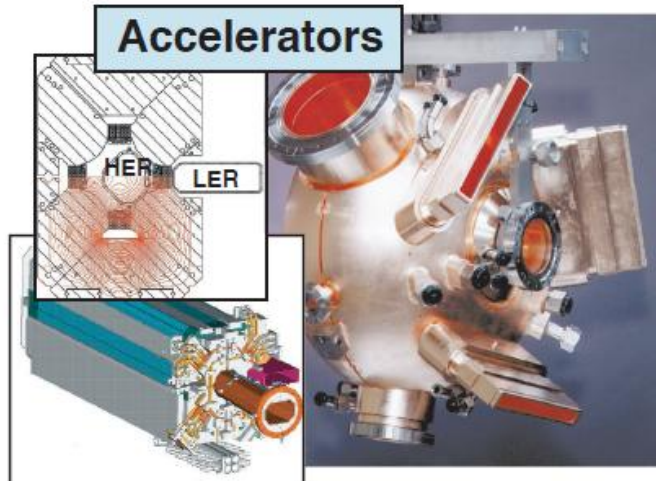
Lawrence Livermore National Laboratory

LLNL-PRES-471096

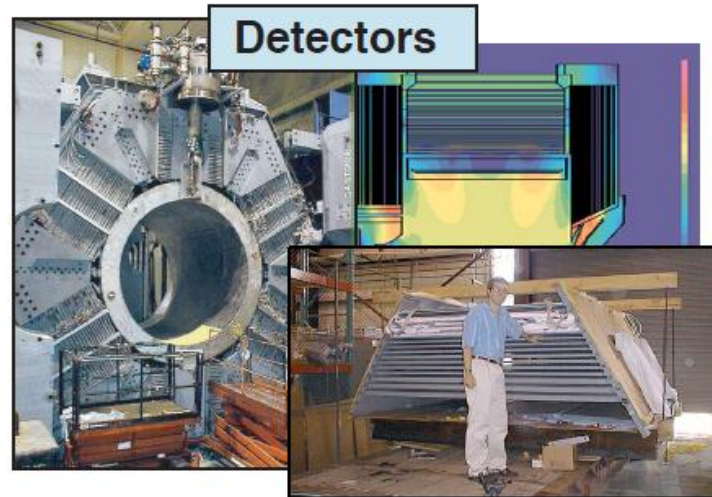
LLNL contributions touch every aspect of the B Factory Project



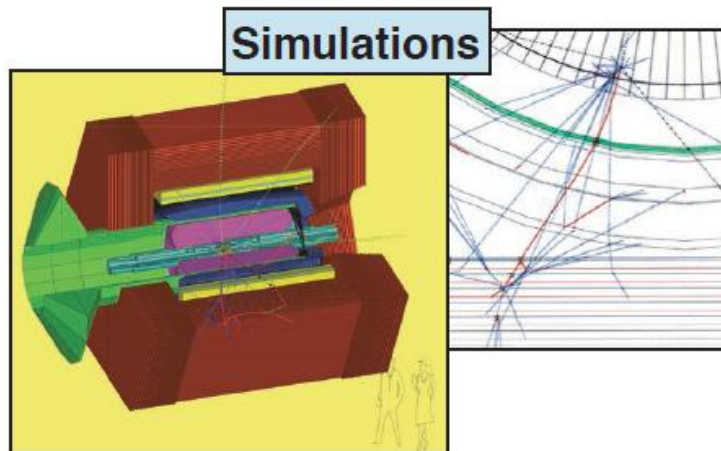
Accelerators



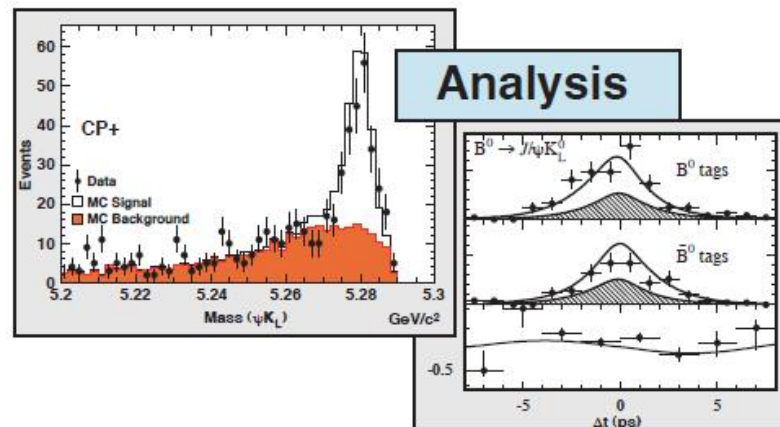
Detectors



Simulations



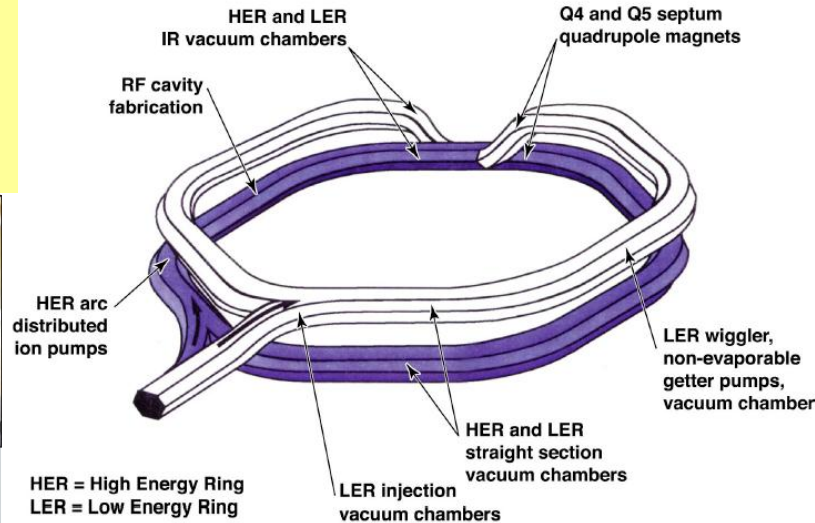
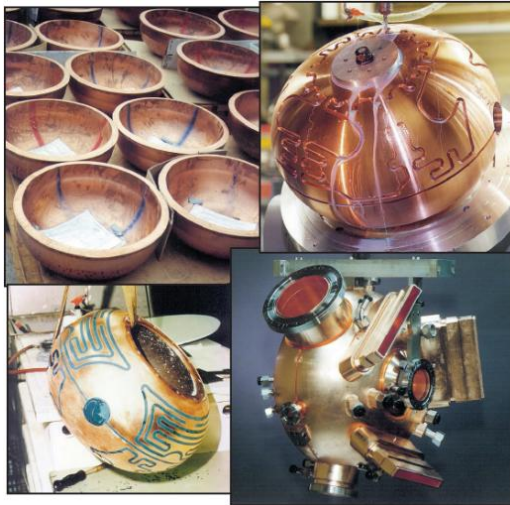
Analysis



Accelerator: LLNL contributed to nearly every major accelerator system in PEP-II



Designed RF cavities and established industry consortium for production

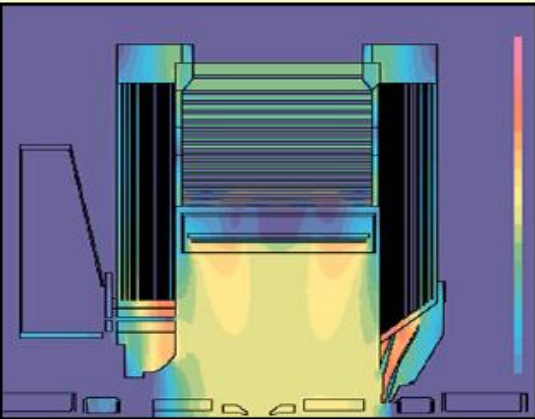


R&D to demonstrate high speed pumping required in HER

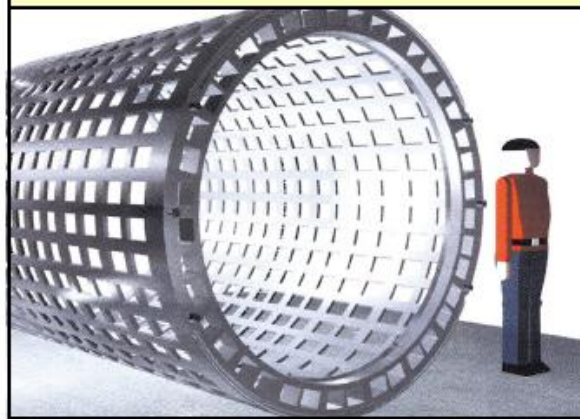


Detector component design and engineering

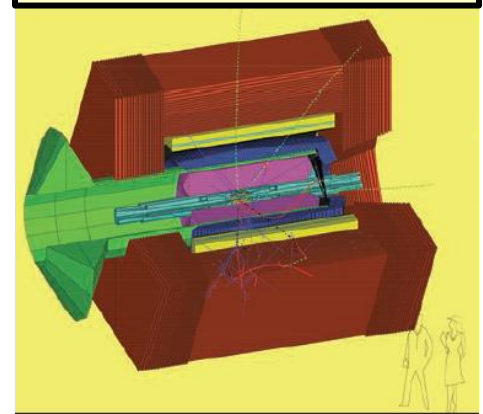
Magnet Engineering



Calorimeter Engineering



Detector Simulation



- Magnet field modeling for flux return
- Solenoid design and construction
- Calorimeter support structure w/ earthquake load
- Built complete GEANT detector model
- Optimized muon and K_L detector configuration
- Built B event generators: BEGET \rightarrow EvtGen

IFR Detector construction and commissioning

RPC assembly, cosmic ray testing and insertion



Cylindrical RPC construction



RPC gas system design and operation



IFR commissioning, RPC upgrade and μ/K_L reconstruction algorithms

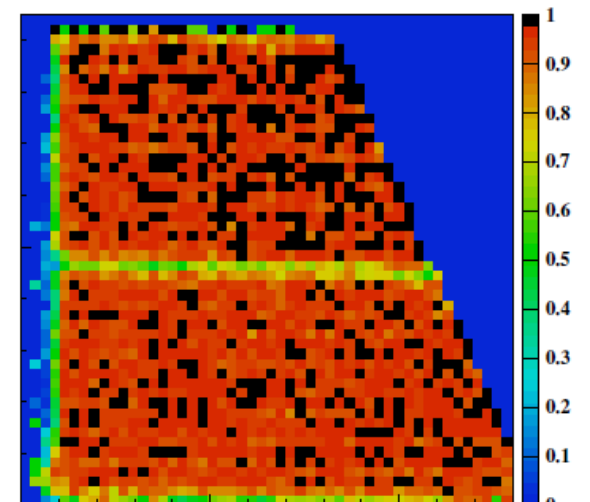
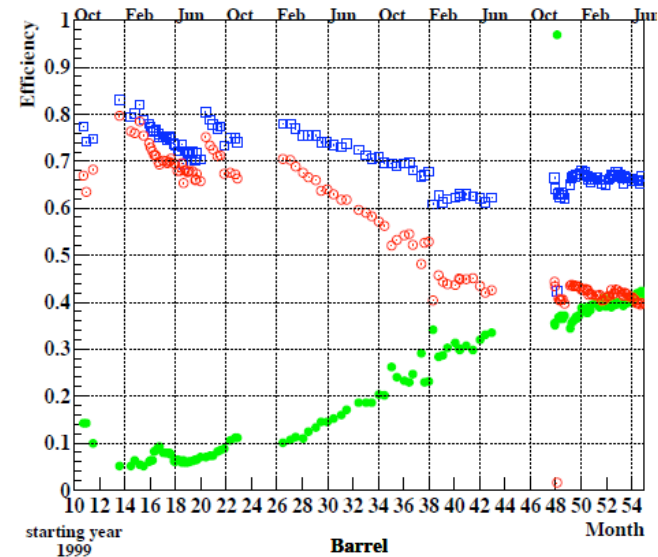


Developed chamber efficiency calculation based on m-pair data

- Automated calibration performed every 12 hours as part of event reconstruction.
- Integrated into μ ID algorithms to optimally account for dead regions

Developed test stand at SLAC for FWD endcap replacement chamber testing

- BABAR DAQ system
- Comprehensive acceptance testing including structural integrity and performance.



LLNL played critical role in $B \rightarrow (cc)K_S \sin 2\beta$ analysis and CP violation discovery in B mesons



K_L reconstruction

- Developed calorimeter K_L ID
- Initial observation of $J/\psi K_L$

Developed one of two CP fitting packages

- Developed flexible background parameterization approach for larger data samples
- Developed direct CP fit procedure

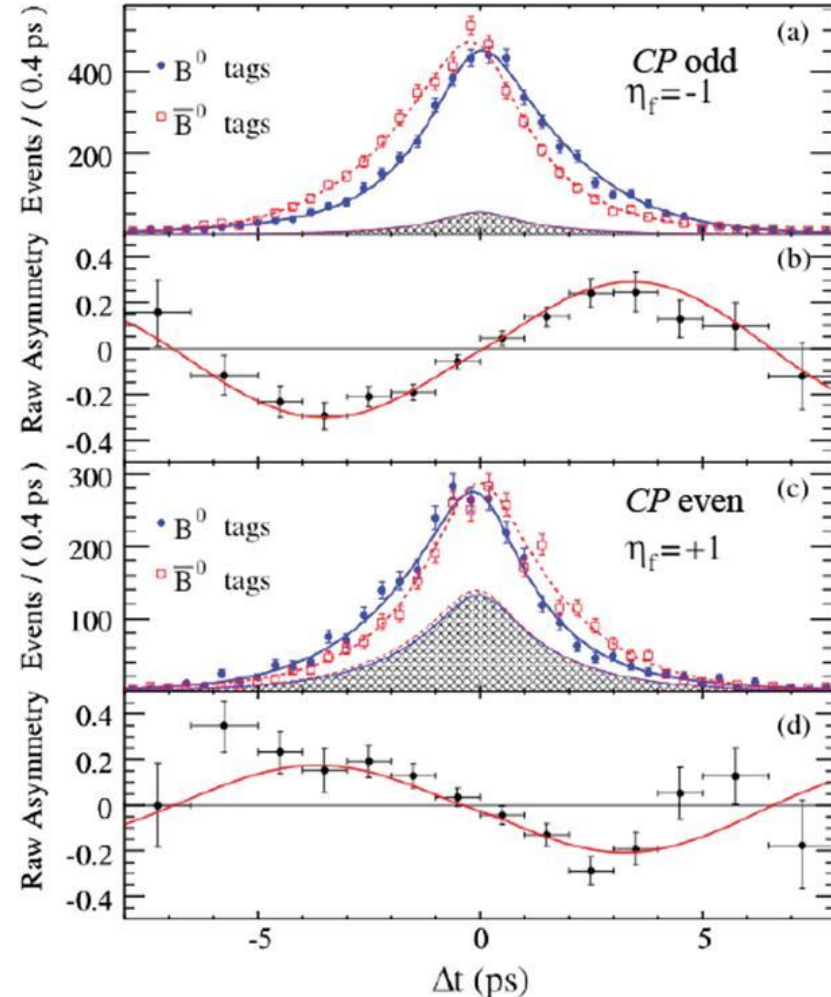
Improved flavor tagging algorithm

- Developed neural network based flavor tag algorithm used in 2004 and later $\sin 2\beta$ analysis.

Analysis leadership

- K_L working group convener (1999-2000)
- Time-dependent analysis working group convener (2003-2004)

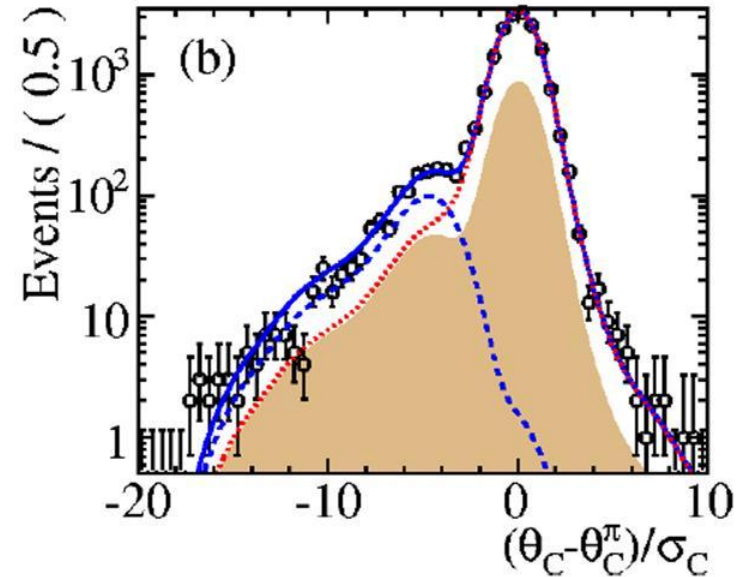
$\sin 2\beta$ results on full data sample



Analyzed $B \rightarrow DK\pi$ for sensitivity to CKM angle γ : 3 body analysis technique was promising method



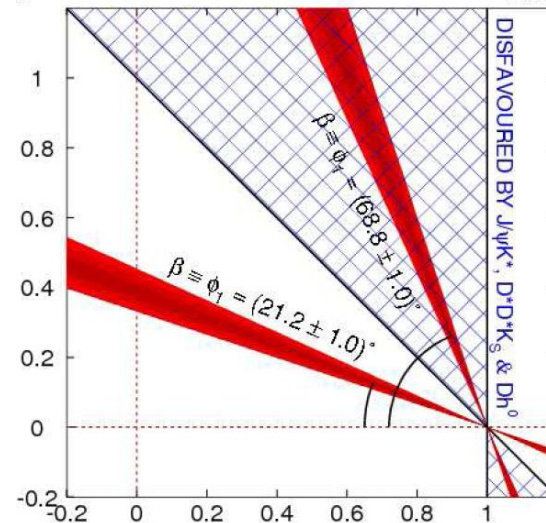
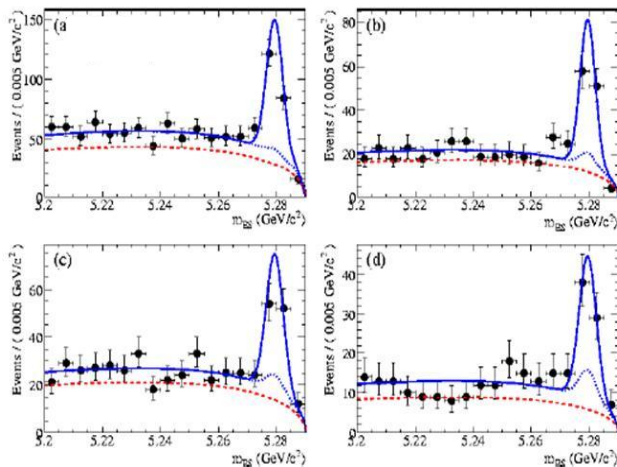
- Sensitivity to γ from interference between interfering amplitudes across Dalitz plot.
- Goal to measure $b \rightarrow u$ component, which is key to γ determination.
- Developed a dramatically improved DIRC θ_c parameterization for $K\pi$ particle ID
- Published results showing much higher than anticipated backgrounds and did not observe the $b \rightarrow u$ amplitude.



Branching Fract.	Result
$\bar{D}^0 K^+ \pi^-$ (no $D^{*-} K^+$)	$(88 \pm 15 \pm 9) \times 10^{-6}$
$\bar{D}^0 K^{*0}$ ($K^{*0} \rightarrow K^+ \pi^-$)	$(38 \pm 6 \pm 4) \times 10^{-6}$
$D_2^{*-} K^+$ ($D_2^{*-} \rightarrow \bar{D}^0 \pi^-$)	$(18.3 \pm 4.0 \pm 3.1) \times 10^{-6}$
Other $\bar{D}^0 K^+ \pi^-$	$(26 \pm 8 \pm 4) \times 10^{-6}$
$D^0 K^+ \pi^-$	$< 19 \times 10^{-6}$
$D^{*-} K^+ / D^{*-} \pi^+$	$(7.76 \pm 0.34 \pm 0.29)\%$

Breaking the β ambiguity: Measure $\cos 2\beta$ using color suppressed decays: $B \rightarrow D^0 \pi^0$, $D^0 \rightarrow K_s \pi^+ \pi^-$

- Published analysis based on one of the first proposed methods to reduce 4 fold ambiguity in β from $\sin 2\beta$ measurement.
- Approach required time-dependent analysis of $K_s \pi \pi$ Dalitz plot.
- Data favored Standard Model solution.



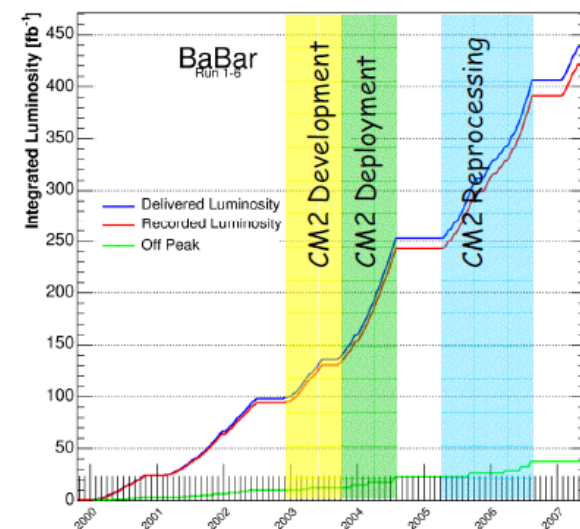
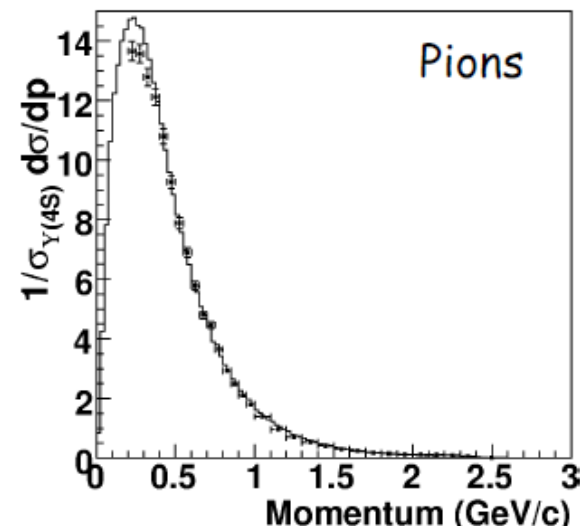
$$\left. \begin{aligned} \cos 2\beta &= 0.54 \pm 0.54 \pm 0.08 \pm 0.18 \\ \sin 2\beta &= 0.45 \pm 0.35 \pm 0.05 \pm 0.07 \\ |\lambda| &= 0.975 \pm 0.09 \pm 0.01 \pm 0.002 \end{aligned} \right\}$$

$\cos 2\beta > 0$ @87% CL

LLNL contributed to all facets of BABAR offline computing



- Co-authored EvtGen package designed for modeling details of complex sequential decays common to B decays
- Developed data sample skimming application and standardized physics analysis package for major revision of BABAR computing model (“CM2”)
- Led reconstruction software and offline project through major data reprocessing and simulation production. Developed data quality procedures



Long history of collaboration leadership

Analysis:

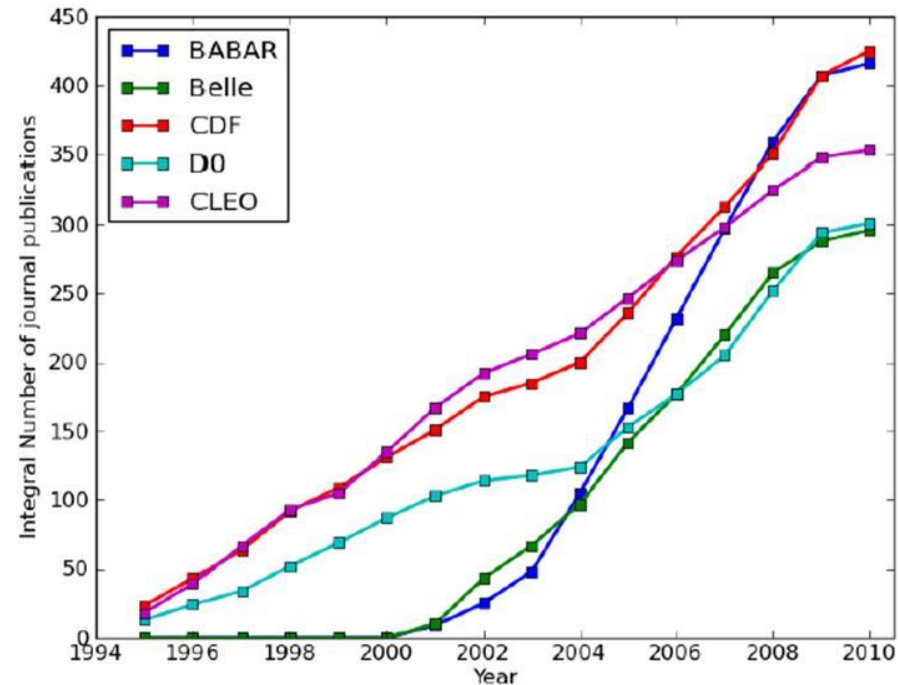
- Publications Board chair
- Time-dependent analysis convener
- Flavor tagging group convener
- K_L working group leader

Detector:

- IFR operations manager
- IFR software manager
- Solenoid project engineer

Offline software/computing:

- Offline coordinator (reconstruction, simulation, skimming, etc)
- Reconstruction manager
- Physics software coordinator





LLNL on BABAR Summary

Analysis phases of BABAR

- 4 postdocs
 - 3 have remained in high-energy physics
[now at LLNL, CalTech, Rudjer Boskovic Inst. (Zagreb)]
 - 4 additional LLNL postdocs during construction phase [LLNL, BNL]
- Direct contributors or sole authors of 9 BABAR physics publications
- 12 conference presentations

Ongoing work:

- Involved in BABAR detector NIM paper update
- Support EvtGen use in BABAR analysis and super-B design.
 - EvtGen is defacto generator code for B-meson decays.
 - LHC-b experiment is pushing current development, however core algorithm expertise is limited to original authors



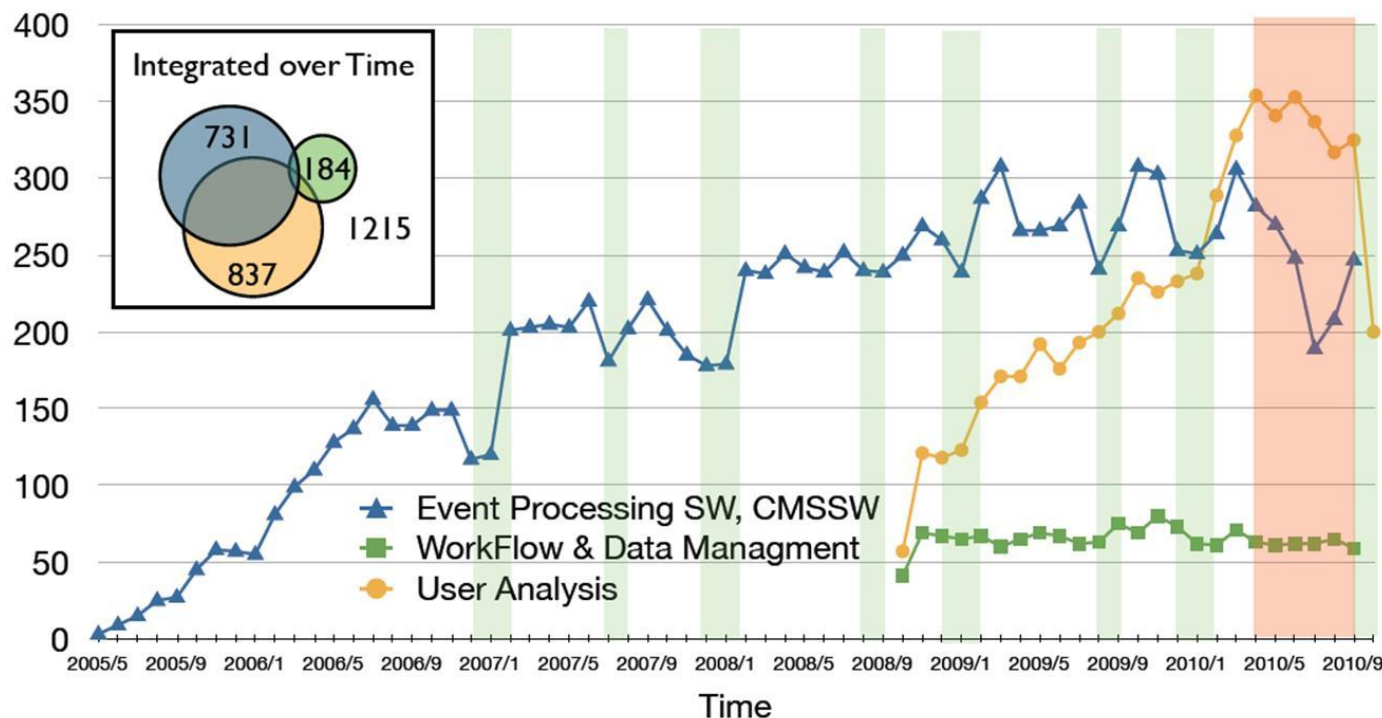
Backups..

CMS Offline software

David J Lange

Lawrence Livermore National Laboratory

Offline software releases are basis of production applications and analysis in CMS



CMS rewrote its offline software starting in 2005 following a new development model. The coherent framework of “CMSSW” required that the release build/distribution tools be revamped at same time

We led the software development tools group during this transition



Development philosophy:

- Build procedure should ensure consistency and flexibility.
- Ensure ability to recover from bugs that prevent CMS from taking data (even if in non-CMS codes).
- Ensure long term ease of maintenance
- Minimize changes to user interfaces

Our approach has paid off:

- A full release build is 4 hours on a single computer. We apply fixes at all levels (gcc, root, CMSSW) quickly and easily
- Most release coordination work turned over to non-experts. Most of their work focuses on ensuring the quality of CMS developer contributions
- We preserved the user primary interface to CMS software while making dramatic performance improvements behind the scenes

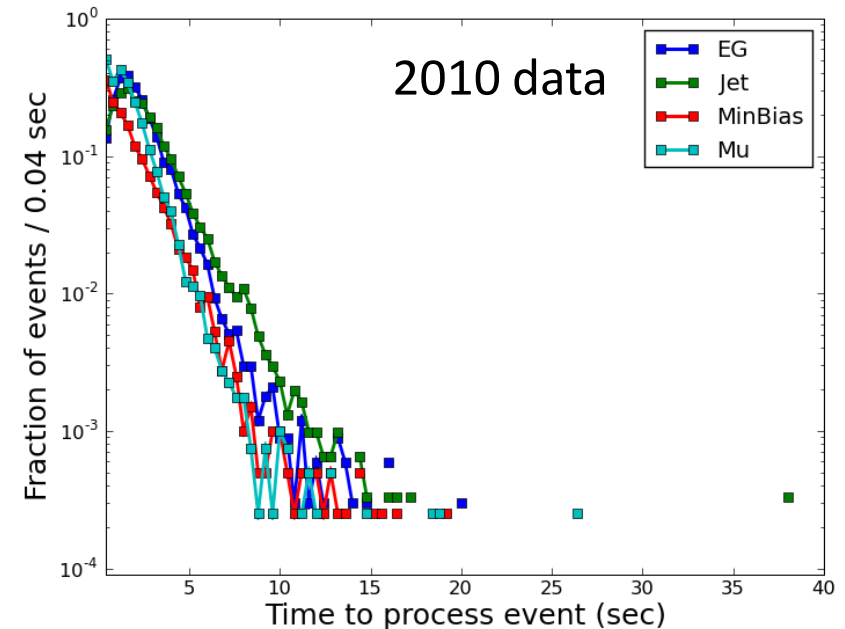
Event reconstruction algorithm performance a success on 2010 data



The reconstruction software is a major undertaking that spans the detector and physics groups in CMS. We are focusing on the overall integration and technical performance

Major milestones on first data:

- Developed effective mechanism to include latest developments into stable production system
- CPU requirement below TDR budget
- Very low job crash rate
- Many analyses use “prompt reconstruction” data for publication

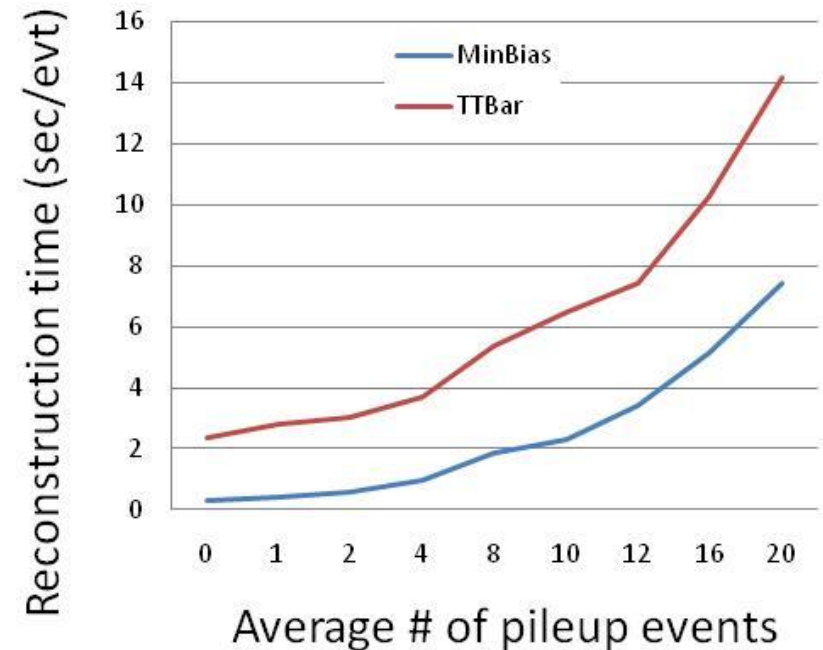


High pileup expected
for 2011 and 2012 brings new
performance challenges

Performance optimization saves computing time and money



- We are part of small group working to improve technical performance of the reconstruction algorithms.
- Tradeoff between time required to reconstruct events vs. computing budget, physics reach and allowable trigger rate.
- Experts pursuing opportunities for improvement has proven to work better than leaving this work to individual developers
- Pileup has big influence on event reconstruction time.
 - Focusing on technical improvement to minimize increases in physics thresholds



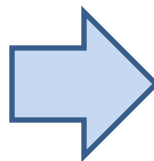
Further ahead: Shared memory approaches needed to use future computing effectively



CPU processor development now driven to reduce wall plug power

Moore's Law increase in
processor speed

Traditional HEP approach:
One job = One CPU core



Evolution to many CPU
cores per machine

New approaches needed to
optimally benefit from CPU power

- With focus on performance, CMS is ahead of this issue. We are deploying first solutions this year. Current and near future computing can be effectively used via 'forking/copy-on-write' mechanism
- LLNL looking at next-generation solutions.
 - Increasing memory sharing in forking implementation
 - Evaluating mechanisms for algorithm level threading. Needed as computing moves from "multi-core" to "many core" platforms.